Hong, H., Kim, Y., Scholten, G. & Sendra, J. R. 2017, “Resultants over commutative idempotent semirings I: Algebraic aspect”, Journal of Symbolic Computation, vol. 79, Part 2, pp. 285-308

Available at https://doi.org/10.1016/j.jsc.2016.02.009

The final journal version of this paper appears in [Hoon Hong, Yonggu Kim, Georgy Scholten, J. Rafael Sendra. Resultants over Commutative Idempotent Semirings I: (Algebraic aspect) Journal of Symbolic Computation 79 (2017) 285-308.] and it is available at Elsevier via http://dx.doi.org/10.1016/j.jsc.2016.02.009

Resultants over Commutative Idempotent Semirings I: (Algebraic aspect)

Hoon Hong, Yonggu Kim, Georgy Scholten, J. Rafael Sendra *

Abstract

The resultant theory plays a crucial role in computational algebra and algebraic geometry. The theory has two aspects: algebraic and geometric. In this paper, we focus on the algebraic aspect. One of the most important and well known algebraic properties of the resultant is that it is equal to the determinant of the Sylvester matrix. In 2008, Odagiri proved that a similar property holds over the tropical semiring if one replaces subtraction with addition. The tropical semiring belongs to a large family of algebraic structures called commutative idempotent semiring. In this paper, we prove that the same property (with subtraction replaced with addition) holds over an *arbitrary* commutative idempotent semiring.

1 Introduction

The main contribution of this paper is adapting a certain important algebraic property of resultant (over commutative rings) to commutative idempotent semirings. The work was inspired by Odagiri's work [30] where the algebraic property of resultant is adapted to a particular commutative idempotent semiring, namely tropical semiring. Below we elaborate on the above statements.

The resultant plays a crucial role in (computational) algebra and algebraic geometry [38, 34, 26, 9]. Let

$$f = (x - \alpha_1) \cdots (x - \alpha_m)$$
$$g = (x - \beta_1) \cdots (x - \beta_n)$$

be two polynomials over a commutative ring. The resultant $\mathbf{R}$ of f and g is defined as

$$\mathbf{R} = \prod_{\substack{1 \le i \le m \\ 1 \le j \le n}} (\alpha_i - \beta_j)$$

and the Sylvester expression of f and g is defined as

$$\mathbf{S} = \det M$$

where M is a certain matrix whose entries are from the coefficients of the two polynomials. One of the most important and well known properties of the resultant [38, 10] is that

$$\mathbf{R} = \mathbf{S}.$$

The *tropical semiring* is a variant of a commutative ring where it is equipped with a total order and that the addition operation is defined as maximum. As a result, it does not allow subtraction (due to lack of

*H. Hong was partially supported by US NSF 1319632. A part of this work was also developed while H. Hong was visiting J.R. Sendra at the *Universidad de Alcalá*, under the frame of the project *Giner de los Rios*. Y. Kim was financially supported by Chonnam National University in the program, 2012. J.R. Sendra belongs to the Research Group ASYNACS (Ref. CCEE2011/R34) and is partially supported by the Spanish Ministerio de Economía y Competitividad under the Project MTM2014-54141.

additive inverse; hence the name semiring). It has been intensively investigated due to numerous interesting applications [35, 33, 32, 18, 3, 15, 36, 19, 8, 37, 25, 5, 27].

There have been several adaptations of the properties of the resultant (over commutative rings) to the tropical semiring [29, 11, 4, 39, 30, 22]. In particular, Odagiri [30] proved that the algebraic property of the resultant still holds if one simply replaces subtraction with addition, that is, if we let

$$f = (x + \alpha_1) \cdots (x + \alpha_m)$$
$$g = (x + \beta_1) \cdots (x + \beta_n)$$

and redefine the resultant as

$$\mathbf{R} = \prod_{\substack{1 \le i \le m \\ 1 \le j \le n}} (\alpha_i + \beta_j),$$

and redefine the Sylvester expression as

$$\mathbf{S} = \operatorname{per} M$$

then

$$\mathbf{R} = \mathbf{S}.$$

In [20] and [21], Izhakian and Rowen introduced a family of algebraic structures called *supertropical semirings* (STS) which generalizes the tropical semiring. Furthermore, they proved that the above algebraic property of resultant holds for an arbitrary STS.[1]

In this paper, we consider a different generalization. The tropical semiring belongs to a large family of algebraic structures called *commutative idempotent semiring* (CIS) [31, 23, 16, 13, 12]. As the name indicates, a commutative idempotent semiring is similar to a commutative ring, except that we do not require subtraction (additive inverse) but we instead require additive idempotency. There are many interesting algebraic structures that are CIS (see Section 2). Many CIS structures are not STS.

Hence one naturally wonders whether we can extend the algebraic property of the resultants on tropical semiring to the whole family of CIS. The main contribution of this paper is to answer affirmatively, proving that the algebraic property of resultant indeed holds for arbitrary CIS, not just for the tropical semiring.

For proving the property, we, at the beginning, naturally attempted to generalize/relax the proof technique of Odagiri. However we found it practically impossible mainly because his proof crucially exploits the fact that the tropical semiring has a total order. Since CIS, in general, does not require a total order, we had to develop a different proof technique. The new technique consists of the following four parts:

1. Represent each term in $\mathbf{R}$ as a certain boolean matrix, which we call a *res*-representation.
2. Represent each term in $\mathbf{S}$ as a certain pair of boolean matrices, which we call a *syl*-representation.
3. Show that if a term has a *res*-representation then it has a *syl*-representation.
4. Show that if a term has a *syl*-representation then it has a *res*-representation.

The representation of terms as boolean matrices are not essential from logical point of view, but they are extremely helpful in discovering, explaining and understanding the steps of the proof. The proof is constructive, that is, it provides an algorithm that takes a *res*-representation and produces a *syl*-representation, and vice versa. The implementation of the algorithms in Maple [28, 2] can be downloaded from

`http://www.math.ncsu.edu/~hong/rcis/`

As mentioned in the abstract, a resultant theory has two important aspects: algebraic and geometric. An algebraic theory investigates relationships between different representations of symmetric polynomials in the roots (treated as indeterminates). A geometric theory investigates the notion of roots and their existence. Both theories are crucial for the development of a complete theory.

[1] See Corollary 4.13. [21]. It requires that the polynomials are tangible. However, it seems that the requirement is not needed for proving the algebraic part of the claim.

Historically (over fields, tropical fields, supertropical fields), an algebraic theory and a geometric theory were often developed and reported separately, mainly because each was non-trivial and also interesting on its own. We find that it is the same for the commutative idempotent semiring case. We observe that its algebraic theory is nontrivial (due to lack of total order) and also interesting on its own (provides algorithms for converting between two representations of a term). Hence we first report an algebraic theory in this paper. We expect that its geometric theory is also non-trivial and interesting on its own. We plan to report a geometric theory in a sequel.

The paper is structured as follows. In Section 2, we recall the axiomatic definition of CIS and list several algebraic structures that satisfy the axioms. In Section 3, we give a precise statement of the main result of this paper. In Section 4, we illustrate the main result on some examples based on the algebraic structures listed in Section 2. In Section 5, we finally provide a detailed and general proof of the main result.

2 Review of Commutative Idempotent Semiring

In this section, we review the definition of commutative idempotent semiring, and list a few examples. For more details, see [23, 17, 12, 14].

Definition 1 (Commutative Idempotent Semiring)**.** *A Commutative Idempotent Semiring (CIS) is a tuple* $(\mathcal{I}, +, \times, 0, 1)$ *where* $\mathcal{I}$ *is a set,* $+$ *and* $\times$ *are binary operations over* $\mathcal{I}$ *and* $0, 1$ *are elements of* $\mathcal{I}$ *such that the following properties hold for all* $a, b, c \in \mathcal{I}$*:*

$+$	$\times$	$+$ *and* $\times$
$a + b = b + a$	$a \times b = b \times a$	$(a + b) \times c = a \times c + b \times c$
$(a + b) + c = a + (b + c)$	$(a \times b) \times c = a \times (b \times c)$	
$a + 0 = a$	$a \times 1 = a$	
$a + a = a$	$a \times 0 = 0$	

Remark 1. *Note that CIS does not require the existence of additive inverse, thus semiring. It instead requires idempotency, thus idempotent semiring.*

Example 1. *We list a few examples of commutative idempotent semirings (CIS).*

$\mathcal{I}$	$+$	$\times$	0	1
$\mathbb{R} \cup \{-\infty\}$ *(tropical CIS)*	*maximum*	*addition*	$-\infty$	0
Power set of a set S	*union*	*intersection*	$\emptyset$	S
Topology on a set S	*union*	*intersection*	$\emptyset$	S
Compact convex subsets of $\mathbb{R}^n$	*convex hull*	*Minkowski sum*	$\emptyset$	$\{0\}$
Sequences over a CIS A	*component-wise*	*convolution*	$(0_A, \ldots)$	$(1_A, 0_A, \ldots)$
Ideals of a commutative ring R *with unity*	*ideal sum*	*ideal product*	$\{0_R\}$	$\langle 1_R \rangle$
Polynomials over a CIS A	*polynomial sum*	*polynomial product*	0_A	1_A

For the sequences,

$$\begin{array}{lcl} \textit{component-wise} & : \forall\, i \geq 0 \quad (u+v)_i & = \quad u_i +_A v_i \\ \textit{convolution} & : \forall\, i \geq 0 \quad (u \times v)_i & = \quad \sum_{A,\ j,k \geq 0, j+k=i} \ u_j \times_A v_k \end{array}$$

3 Main Result

In this section, we give a precise statement of the main result of this paper and, in the next section, we illustrate it on a couple of examples. Let $\mathcal{I}$ be a CIS. Let $\alpha_1, \ldots, \alpha_m, \beta_1, \ldots, \beta_n \in \mathcal{I}$. Let

$$f = \prod_{i=1}^{m} (x + \alpha_i) \quad \in \mathcal{I}[x], \qquad g = \prod_{j=1}^{n} (x + \beta_j) \quad \in \mathcal{I}[x].$$

Definition 2 (Resultant)**.** *The resultant* $\mathbf{R}$ *of* f *and* g *is defined as*

$$\mathbf{R} = \prod_{\substack{1 \le i \le m \\ 1 \le j \le n}} (\alpha_i + \beta_j) \quad \in \mathcal{I}.$$

Definition 3 (Sylvester Matrix)**.** *Let* $a_0, \ldots, a_m \in \mathcal{I}$ *and* $b_0, \ldots, b_n \in \mathcal{I}$ *be such that*

$$f = \sum_{i=0}^{m} a_{m-i} x^i, \qquad g = \sum_{j=0}^{n} b_{n-j} x^j.$$

Then the Sylvester matrix of f *and* g *is defined as*

$$M = \underbrace{\begin{bmatrix} a_0 & \cdots & \cdots & \cdots & a_m & & \\ & \ddots & & & & \ddots & \\ & & a_0 & \cdots & \cdots & \cdots & a_m \\ b_0 & \cdots & \cdots & b_n & & & \\ & \ddots & & & \ddots & & \\ & & \ddots & & & \ddots & \\ & & & b_0 & \cdots & \cdots & b_n \end{bmatrix}}_{m+n} \begin{matrix} \left.\vphantom{\begin{matrix} a \\ a \\ a \end{matrix}}\right\} n \\ \left.\vphantom{\begin{matrix} a \\ a \\ a \end{matrix}}\right\} m \end{matrix}$$

Definition 4 (Sylvester expression)**.** *The Sylvester expression* $\mathbf{S}$ *of* f *and* g *is defined as the permanent of the Sylvester matrix of* f *and* g*, that is,*

$$\mathbf{S} = \operatorname{per}(M) \quad \in \mathcal{I}.$$

Theorem 1 (Main Result)**.** $\mathbf{R} = \mathbf{S}$.

Example 2. *We illustrate the "meaning" of the main result for the two polynomials of degrees 3 and 2*

$$f = (x + \alpha_1)(x + \alpha_2)(x + \alpha_3), \qquad g = (x + \beta_1)(x + \beta_2).$$

Then the resultant $\mathbf{R}$ *of* f *and* g *is given by*

$$\mathbf{R} = (\alpha_1 + \beta_1)(\alpha_1 + \beta_2)(\alpha_2 + \beta_1)(\alpha_2 + \beta_2)(\alpha_3 + \beta_1)(\alpha_3 + \beta_2).$$

Expanding f *and* g *we get*

$$f = a_0 x^3 + a_1 x^2 + a_2 x^1 + a_3 x^0, \quad g = b_0 x^2 + b_1 x + b_0 x^0$$

where

$$\begin{array}{ll} a_0 = 1, & b_0 = 1, \\ a_1 = \alpha_1 + \alpha_2 + \alpha_3, & b_1 = \beta_1 + \beta_2, \\ a_2 = \alpha_1\alpha_2 + \alpha_1\alpha_3 + \alpha_2\alpha_3, & b_2 = \beta_1\beta_2, \\ a_3 = \alpha_1\alpha_2\alpha_3. & \end{array}$$

Thus the Sylvester expression $\mathbf{S}$ *of* f *and* g *is given by*

$$\mathbf{S} = \text{per}\begin{bmatrix} 1 & \alpha_1+\alpha_2+\alpha_3 & \alpha_1\alpha_2+\alpha_1\alpha_3+\alpha_2\alpha_3 & \alpha_1\alpha_2\alpha_3 & \\ & 1 & \alpha_1+\alpha_2+\alpha_3 & \alpha_1\alpha_2+\alpha_1\alpha_3+\alpha_2\alpha_3 & \alpha_1\alpha_2\alpha_3 \\ 1 & \beta_1+\beta_2 & \beta_1\beta_2 & & \\ & 1 & \beta_1+\beta_2 & \beta_1\beta_2 & \\ & & 1 & \beta_1+\beta_2 & \beta_1\beta_2 \end{bmatrix}$$

When we expand $\mathbf{R}$ *and* $\mathbf{S}$*, we observe that* $\mathbf{R}$ *and* $\mathbf{S}$ *have the same terms, but some terms appear a different number of times. For example,* ${\alpha_1}^2\alpha_2\alpha_3\beta_1\beta_2$ *appears two times in* $\mathbf{R}$ *and six times in* $\mathbf{S}$*. However, since a commutative idempotent semiring ignores additive multiplicities, it does not matter. Hence we see that* $\mathbf{R} = \mathbf{S}$*.*

4 Examples

In this section, we will show computational examples on some of the *particular* CIS structures given in Example 1 of Section 2 to confirm the validity of the main result (Theorem 1) before its general proof (given in Section 5). We will use *structure-specific languages* whenever possible, in the hope that it would demonstrate the applicability of the main result in apparently very different contexts. We will confirm the main result via direct structure-specific computations. For easy computation, we consider only degree two polynomials.

Example 3 (Power set)**.** *Let* $\mathcal{I}$ *be the CIS of the power set of* $\mathbb{R}$*. Consider*

$$f = (x \cup [1,2]) \cap (x \cup [3,4]), \quad g = (x \cup [2,3]) \cap (x \cup [4,5])$$

We show that $\mathbf{R} = \mathbf{S}$ *for the above* f *and* g*, via direct computations. Note*

$$\begin{aligned} \mathbf{R} &= ([1,2]\cup[2,3]) \cap ([1,2]\cup[4,5]) \cap ([3,4]\cup[2,3]) \cap ([3,4]\cup[4,5]) \\ &= [1,3] \cap ([1,2]\cup[4,5]) \cap [2,4] \cap [3,5] \\ &= [1,3] \cap ([1,2]\cup[4,5]) \cap [3,4] \\ &= [3,3] \cap ([1,2]\cup[4,5]) \\ &= \emptyset \end{aligned}$$

$$\begin{aligned} \mathbf{S} &= \text{per}\begin{bmatrix} \mathbb{R} & [1,2]\cup[3,4] & [1,2]\cap[3,4] & \emptyset \\ \emptyset & \mathbb{R} & [1,2]\cup[3,4] & [1,2]\cap[3,4] \\ \mathbb{R} & [2,3]\cup[4,5] & [2,3]\cap[4,5] & \emptyset \\ \emptyset & \mathbb{R} & [2,3]\cup[4,5] & [2,3]\cap[4,5] \end{bmatrix} \\ &= \text{per}\begin{bmatrix} \mathbb{R} & [1,2]\cup[3,4] & \emptyset & \emptyset \\ \emptyset & \mathbb{R} & [1,2]\cup[3,4] & \emptyset \\ \mathbb{R} & [2,3]\cup[4,5] & \emptyset & \emptyset \\ \emptyset & \mathbb{R} & [2,3]\cup[4,5] & \emptyset \end{bmatrix} \\ &= \emptyset \end{aligned}$$

Thus we have $\mathbf{R} = \mathbf{S}$*.*

Example 4 (Ideals)**.** *Let* $\mathcal{I}$ *be the CIS of the set of all the ideals of* $\mathbb{C}[v_1, v_2]$*. Consider*

$$f = \Big(x + \langle v_1^2 + v_2^2 - 1^2\rangle\Big)\Big(x + \langle v_1^2 + v_2^2 - 2^2\rangle\Big), \quad g = \Big(x + \langle v_1^2 - v_2^2 - 1^2\rangle\Big)\Big(x + \langle v_1^2 - v_2^2 - 2^2\rangle\Big)$$

In order to simplify the presentation, we will use the following short-hands.

$$I_i = \langle v_1^2 + v_2^2 - i^2 \rangle, \quad J_j = \langle v_1^2 - v_2^2 - j^2 \rangle$$

Then we can write f and g succinctly as

$$f = (x + I_1)(x + I_2), \quad g = (x + J_1)(x + J_2)$$

We show that $\mathbf{R} = \mathbf{S}$ for the particular f and g, via direct computations. Note

$$\begin{aligned}
\mathbf{R} &= (I_1 + J_1)\,(I_1 + J_2)\,(I_2 + J_1)\,(I_2 + J_2) \\
\\
\mathbf{S} &= \text{per} \begin{bmatrix} \langle 1 \rangle & I_1 + I_2 & I_1 I_2 & \{0\} \\ \{0\} & \langle 1 \rangle & I_1 + I_2 & I_1 I_2 \\ \langle 1 \rangle & J_1 + J_2 & J_1 J_2 & \{0\} \\ \{0\} & \langle 1 \rangle & J_1 + J_2 & J_1 J_2 \end{bmatrix} \\
&= \text{per} \begin{bmatrix} \langle 1 \rangle & \langle 1 \rangle & I_1 I_2 & \{0\} \\ \{0\} & \langle 1 \rangle & \langle 1 \rangle & I_1 I_2 \\ \langle 1 \rangle & \langle 1 \rangle & J_1 J_2 & \{0\} \\ \{0\} & \langle 1 \rangle & \langle 1 \rangle & J_1 J_2 \end{bmatrix} \\
&= (I_1 I_2)^2 + (J_1 J_2)^2 + (I_1 I_2)(J_1 J_2) + I_1 I_2 + J_1 J_2
\end{aligned}$$

After carrying out ideal additions and multiplications in a straightforward manner (see e.g. [1, 9, 24]), we obtain the following sets of generators for $\mathbf{R}$ and for $\mathbf{S}$.

$\mathbf{R} = \langle {v_1}^8 - 2\,{v_1}^4{v_2}^4 + {v_2}^8 - 10\,{v_1}^6 + 10\,{v_1}^2{v_2}^4 + 33\,{v_1}^4 - 17\,{v_2}^4 - 40\,{v_1}^2 + 16, \ldots 14 \text{ more polynomials} \ldots \rangle$

$\mathbf{S} = \langle {v_1}^4 - 2\,{v_1}^2{v_2}^2 + {v_2}^4 - 5\,{v_1}^2 + 5\,{v_2}^2 + 4, \ldots 4 \text{ more polynomials} \ldots \rangle$

Note that the generators of the two ideals look very different. However, after computing the reduced Gröbner basis (see e.g. [6, 7, 9]) of the generators with respect to the total degree order ($v_1 \succ v_2$), we obtain

$$\begin{aligned}
\mathbf{R} &= \langle 2\,{v_1}^2{v_2}^2 - 5\,{v_2}^2, \;\; {v_1}^4 + {v_2}^4 - 5\,{v_1}^2 + 4, \;\; 4\,{v_2}^6 - 9\,{v_2}^2 \rangle \\
\mathbf{S} &= \langle 2\,{v_1}^2{v_2}^2 - 5\,{v_2}^2, \;\; {v_1}^4 + {v_2}^4 - 5\,{v_1}^2 + 4, \;\; 4\,{v_2}^6 - 9\,{v_2}^2 \rangle
\end{aligned}$$

Thus we have $\mathbf{R} = \mathbf{S}$.

5 Proof of Main Result

In this section, we provide a proof for the main result. Recall that the main theorem (Theorem 1) claims that $\mathbf{R} = \mathbf{S}$, that is, a term appears in $\mathbf{R}$ if and only if it appears in $\mathbf{S}$. It follows immediately from the following four lemmas.

Lemma 2: A term occurs in $\mathbf{R}$ iff it has a res-representation (a certain boolean matrix).

Lemma 3: A term occurs in $\mathbf{S}$ iff it has a syl-representation (a certain pair of boolean matrices).

Lemma 7: If a term has a res-representation then it has a syl-representation.

Lemma 10: If a term has a syl-representation then it has a res-representation.

We will devote one subsection for each lemma. Each subsection ends with the proof of each of the above lemmas. Lemmas 7 and 10 are proved constructively, by providing algorithms (Algorithms 1 and 4) that produce one representation from the other. These algorithms are based on a key lemma (Lemma 6) that establishes a crucial relationship between the two representations (syl and res). Before stating and proving the above lemmas, we introduce notations that will be used throughout this section.

Notation 5. *Let* $M \in \{0,1\}^{m\times n}$.

1. *The* complement *of* M, *written as* $\bar{M}$, *is the* $m \times n$ *matrix defined by*

$$\bar{M}_{ij} = 1 - M_{ij}$$

2. *The* row sum *of* M, *written as* $rs(M)$, *is the* m*-dimensional vector defined by*

$$rs(M) = \left(\sum_{j=1}^{n} M_{ij}\right)_{i=1,\ldots,m}$$

3. *The* column sum *of* M, *written as* $cs(M)$, *is the* n*-dimensional vector defined by*

$$cs(M) = \left(\sum_{i=1}^{m} M_{ij}\right)_{j=1,\ldots,n}$$

4. *The* adjusted row sum *of* M, *written as* $ars(M)$, *is the* m*-dimensional vector defined by*

$$ars(M) = \left(i + \sum_{j=1}^{n} M_{ij}\right)_{i=1,\ldots,m}$$

5. *The* adjusted column sum *of* M, *written as* $acs(M)$, *is the* n*-dimensional vector defined by*

$$acs(M) = \left(j + \sum_{i=1}^{m} M_{ij}\right)_{j=1,\ldots,n}$$

Example 5. *Let*

$$M = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

Then

$$\begin{aligned} rs(M) &= (3,2,2,1) \\ cs(M) &= (2,3,3) \\ ars(M) &= (4,4,5,5) \\ acs(M) &= (3,5,6) \end{aligned}$$

5.1 A term occurs in R iff it has a res-representation.

Definition 6 (Res-representation)**.** *Let* $t = \alpha^{\mu}\beta^{\nu}$ *be a term, where* $\mu \in \mathbb{Z}^m_{\geq 0}$ *and* $\nu \in \mathbb{Z}^n_{\geq 0}$. *Let* $\mathcal{R} \in \{0,1\}^{m\times n}$. *We say that* $\mathcal{R}$ *is a res-representation of* t *if*

- $rs(\mathcal{R}) = \mu$.
- $cs\left(\bar{\mathcal{R}}\right) = \nu$. *Equivalently* $cs(\mathcal{R}) = \bar{\nu}$ *where* $\bar{\nu}_j = m - \nu_j$.

Example 6. *Let $m = 3$ and $n = 2$. Let $t = \alpha_1^2\alpha_2^1\alpha_3^1\beta_1^1\beta_2^1 = \alpha^{(2,1,1)}\beta^{(1,1)}$. Let*

$$\mathcal{R} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

We would like to know whether $\mathcal{R}$ is a res-representation of t. Note

$$\mathcal{R} = \left[\begin{array}{cc||c} 1 & 1 & 2 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \\ \hline\hline & & \mu_i \\ 2 & 2 & \bar{\nu}_j \\ 1 & 1 & \nu_j \end{array}\right]$$

Hence $\mathcal{R}$ is a res-representation of t. Likewise, the following matrix is also a res-representation of t.

$$\mathcal{R} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}$$

Lemma 2. *A term occurs in $\mathbf{R}$ iff it has a res-representation.*

Proof. Recall

$$\mathbf{R} = \prod_{i=1}^{m}\prod_{j=1}^{n}(\alpha_i + \beta_j)$$

Note

$$\begin{aligned}
\mathbf{R} &= \prod_{i=1}^{m}\prod_{j=1}^{n}\sum_{e\in\{0,1\}}\alpha_i^e\beta_j^{\bar{e}} \qquad \text{where } \bar{e} = 1 - e \\
&= \sum_{\mathcal{R}\in\{0,1\}^{m\times n}}\prod_{i=1}^{m}\prod_{j=1}^{n}\alpha_i^{\mathcal{R}_{ij}}\beta_j^{\bar{\mathcal{R}}_{ij}} \\
&= \sum_{\mathcal{R}\in\{0,1\}^{m\times n}}\prod_{i=1}^{m}\prod_{j=1}^{n}\alpha_i^{\mathcal{R}_{ij}}\prod_{i=1}^{m}\prod_{j=1}^{n}\beta_j^{\bar{\mathcal{R}}_{ij}} \\
&= \sum_{\mathcal{R}\in\{0,1\}^{m\times n}}\prod_{i=1}^{m}\alpha_i^{\sum_{j=1}^{n}\mathcal{R}_{ij}}\prod_{j=1}^{n}\beta_j^{\sum_{i=1}^{m}\bar{\mathcal{R}}_{ij}} \\
&= \sum_{\mathcal{R}\in\{0,1\}^{m\times n}}\alpha^{\mu}\beta^{\nu} \text{ where } \mu = rs(\mathcal{R}) \text{ and } \nu = cs\left(\bar{\mathcal{R}}\right)
\end{aligned}$$

The claim follows immediately. □

5.2 A term occurs in S iff it has a syl-representation.

Definition 7 (Properly coupled)**.** *Let $\mathcal{S}_1, \mathcal{S}_2 \in \{0,1\}^{m\times n}$. We say that $\mathcal{S}_1$ and $\mathcal{S}_2$ are properly coupled and write as $PC(\mathcal{S}_1, \mathcal{S}_2)$ iff*

$$\{c'_1, \ldots, c'_n, r'_1, \ldots, r'_m\} = \{1, \ldots, m+n\}$$

where $c' = acs(\mathcal{S}_1)$ and $r' = ars(\mathcal{S}_2)$.

Example 7. *Let*

$$\mathcal{S}_1 = \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} \quad \mathcal{S}_2 = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Since

$$c' = acs\left(\mathcal{S}_1\right) = (2, 5)$$
$$r' = ars\left(\mathcal{S}_2\right) = (1, 3, 4)$$

we have

$$\{2, 5, 1, 3, 4\} = \{1, 2, 3, 4, 5\}$$

Hence we have $PC(\mathcal{S}_1, \mathcal{S}_2)$.

Definition 8 (Syl-representation)**.** *Let* $t = \alpha^{\mu}\beta^{\nu}$ *be a term. Let* $\mathcal{S}_1, \mathcal{S}_2 \in \{0,1\}^{m \times n}$. *We say that* $(\mathcal{S}_1, \mathcal{S}_2)$ *is a syl-representation of* t *if*

- $rs\left(\mathcal{S}_1\right) = \mu$
- $cs\left(\mathcal{S}_2\right) = \nu$
- $PC\left(\mathcal{S}_1, \mathcal{S}_2\right)$

Example 8. *Let* $m = 3$ *and* $n = 2$. *Let* $t = \alpha_1^2\alpha_2^1\alpha_3^1\beta_1^1\beta_2^1 = \alpha^{(2,1,1)}\beta^{(1,1)}$. *Let*

$$\mathcal{S}_1 = \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} \quad \mathcal{S}_2 = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

We would like to know whether $(\mathcal{S}_1, \mathcal{S}_2)$ *is a syl-representation of* t. *Note*

$$\mathcal{S}_1 = \left[\begin{array}{cc||c} 1 & 1 & 2 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \\ \hline\hline & & \mu_i \\ \hline 1 & 3 & c_j \\ 1 & 2 & j \\ 2 & 5 & c_j + j \end{array}\right] \quad \mathcal{S}_2 = \left[\begin{array}{cc||ccc} 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 2 & 3 \\ 0 & 1 & 1 & 3 & 4 \\ \hline\hline & & r_i & i & r_i + i \\ \hline 1 & 1 & \nu_j & & \\ & & & & \\ & & & & \end{array}\right]$$

Hence $(\mathcal{S}_1, \mathcal{S}_2)$ *is a syl-representation of* t. *Likewise, the following pairs of matrices are also syl-representations of* t.

$$\mathcal{S}_1 = \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} \quad \mathcal{S}_2 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}$$

$$\mathcal{S}_1 = \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} \quad \mathcal{S}_2 = \begin{bmatrix} 0 & 0 \\ 1 & 1 \\ 0 & 0 \end{bmatrix}$$

$$\mathcal{S}_1 = \begin{bmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \quad \mathcal{S}_2 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 1 \end{bmatrix}$$

$$\mathcal{S}_1 = \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} \quad \mathcal{S}_2 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 1 \end{bmatrix}$$

$$\mathcal{S}_1 = \begin{bmatrix} 1 & 1 \\ 1 & 0 \\ 1 & 0 \end{bmatrix} \quad \mathcal{S}_2 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 1 \end{bmatrix}$$

Lemma 3. *A term occurs in* $\mathbf{S}$ *iff it has a syl-representation.*

Proof. Let M be the Sylvester matrix of f and g (see Definition 3). Then

$$\begin{aligned}
\mathbf{S} &= \operatorname{per}(M) \\
&= \sum_{\{\sigma_1,\ldots,\sigma_{n+m}\}=\{1,\ldots,n+m\}} \prod_{i=1}^{n+m} M_{i\sigma_i} \\
&= \sum_{\{\sigma_1,\ldots,\sigma_n,\sigma_{n+1},\ldots,\sigma_{n+m}\}=\{1,\ldots,n+m\}} \left(\prod_{j=1}^{n} M_{j\sigma_j}\right)\left(\prod_{i=1}^{m} M_{(n+i)\sigma_{n+i}}\right) \\
&= \sum_{\{\sigma_1,\ldots,\sigma_n,\sigma_{n+1},\ldots,\sigma_{n+m}\}=\{1,\ldots,n+m\}} \left(\prod_{j=1}^{n} a_{\sigma_j-j}\right)\left(\prod_{i=1}^{m} b_{\sigma_{n+i}-i}\right) \\
&= \sum_{\{c_1+1,\ldots,c_n+n,r_1+1,\ldots,r_m+m\}=\{1,\ldots,n+m\}} \left(\prod_{j=1}^{n} a_{c_j}\right)\left(\prod_{i=1}^{m} b_{r_i}\right), \quad \text{by reindexing with } c_j=\sigma_j-j \text{ and } r_j=\sigma_{n+i}-i \\
&= \sum_{\{c_1+1,\ldots,c_n+n,r_1+1,\ldots,r_m+m\}=\{1,\ldots,n+m\}} \left(\prod_{j=1}^{n} \sum_{\substack{S_1\in\{0,1\}^n \\ \sum_{k=1}^{n} S_{1,k}=c_j}} \prod_{i=1}^{m} \alpha_i^{S_{1,i}}\right)\left(\prod_{i=1}^{m} \sum_{\substack{S_2\in\{0,1\}^n \\ \sum_{k=1}^{m} S_{2,k}=r_j}} \prod_{j=1}^{n} \beta_j^{S_{2,j}}\right) \\
&= \sum_{\{c_1+1,\ldots,c_n+n,r_1+1,\ldots,r_m+m\}=\{1,\ldots,n+m\}} \left(\sum_{\substack{\mathcal{S}_1\in\{0,1\}^{m\times n} \\ cs(\mathcal{S}_1)=c}} \prod_{i=1}^{m}\prod_{j=1}^{n} \alpha_i^{\mathcal{S}_{1,i,j}}\right)\left(\sum_{\substack{\mathcal{S}_2\in\{0,1\}^{m\times n} \\ rs(\mathcal{S}_2)=r}} \prod_{i=1}^{m}\prod_{j=1}^{n} \beta_j^{\mathcal{S}_{2,i,j}}\right) \\
&= \sum_{\substack{\mathcal{S}_1,\mathcal{S}_2\in\{0,1\}^{m\times n} \\ cs(\mathcal{S}_1)=c \\ rs(\mathcal{S}_2)=r \\ \{c_1+1,\ldots,c_n+n,r_1+1,\ldots,r_m+m\}=\{1,\ldots,n+m\}}} \prod_{i=1}^{m}\prod_{j=1}^{n} \alpha_i^{\mathcal{S}_{1,i,j}} \prod_{i=1}^{m}\prod_{j=1}^{n} \beta_j^{\mathcal{S}_{2,i,j}} \\
&= \sum_{\substack{\mathcal{S}_1,\mathcal{S}_2\in\{0,1\}^{m\times n} \\ PC(\mathcal{S}_1,\mathcal{S}_2)}} \prod_{i=1}^{m}\prod_{j=1}^{n} \alpha_i^{\mathcal{S}_{1,i,j}} \prod_{j=1}^{n}\prod_{i=1}^{m} \beta_j^{\mathcal{S}_{2,i,j}} \\
&= \sum_{\substack{\mathcal{S}_1,\mathcal{S}_2\in\{0,1\}^{m\times n} \\ PC(\mathcal{S}_1,\mathcal{S}_2)}} \prod_{i=1}^{m} \alpha_i^{\sum_{j=1}^{n} \mathcal{S}_{1,i,j}} \prod_{j=1}^{n} \beta_j^{\sum_{i=1}^{m} \mathcal{S}_{2,i,j}} \\
&= \sum_{\substack{\mathcal{S}_1,\mathcal{S}_2\in\{0,1\}^{m\times n} \\ PC(\mathcal{S}_1,\mathcal{S}_2)}} \alpha^{\mu}\beta^{\nu} \quad \text{where } \mu = rs(\mathcal{S}_1) \text{ and } \nu = cs(\mathcal{S}_2)
\end{aligned}$$

The claim follows immediately. □

5.3 If a term has a res-representation then it has a syl-representation.

The proof is constructive, that is, it provides an algorithm that takes a res-representation and produces a syl-representation (Algorithm 1). The algorithm is immediate from a key lemma (Lemma 6) that establishes

a crucial relationship between the two representations (syl and res). Thus most of this subsection will be devoted in stating and proving the key lemma.

Note that $\mathbf{R}$ is a symmetric expression in $\alpha_1, \ldots, \alpha_m$ and in $\beta_1, \ldots, \beta_n$. Thus for any term in $\mathbf{R}$, every term obtained by permuting $\alpha_1, \ldots, \alpha_m$ and permuting $\beta_1, \ldots, \beta_n$ is also in $\mathbf{R}$. The same holds for $\mathbf{S}$ too. Hence, without loss of generality, we will restrict the proof to the terms $\alpha_1^{\mu_1} \cdots \alpha_m^{\mu_m} \beta_1^{\nu_1} \cdots \beta_n^{\nu_n}$ where μ and ν are in non-increasing order, that is, $\mu_1 \geq \mu_2 \geq \cdots \geq \mu_m$ and $\nu_1 \geq \nu_2 \geq \cdots \geq \nu_n$.

Definition 9 (Bottom-left flushed)**.** *A matrix is called* bottom–left flushed *if all the non-zero entries are flushed to the bottom–left. Let $c \in \{0, \ldots, m\}^n$. Then the* bottom–left flushed *matrix of c, written as $F_c \in \{0,1\}^{m \times n}$, is the* bottom–left flushed matrix *such that $cs(F_c) = c$.*

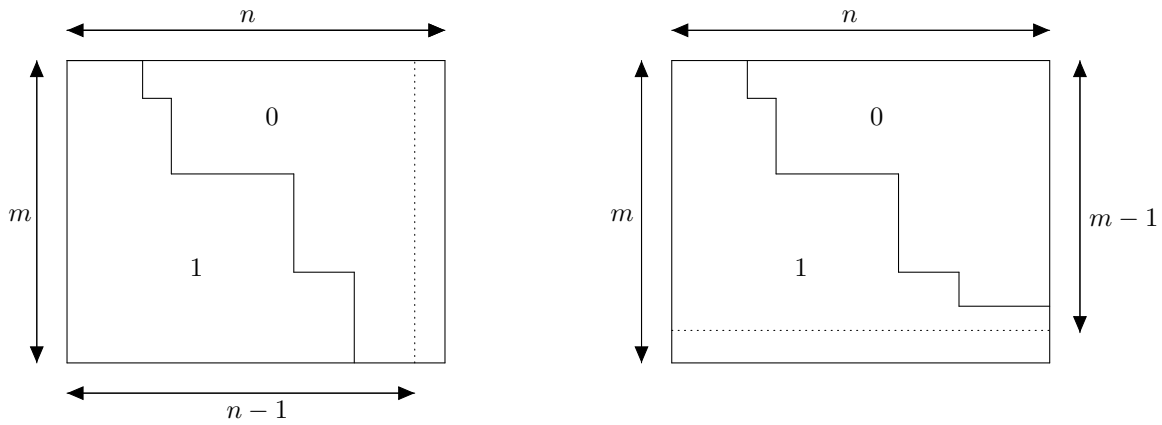
Example 9. *Let*

$$M = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 \end{bmatrix}$$

Then M is flushed. Note also that $M = F_{(4,2,2,1,0)}$.

Lemma 4. *Let $M \in \{0,1\}^{m \times n}$ be bottom-left flushed. Then we have $PC\left(\bar{M}, M\right)$.*

Proof. We will prove by mathematical induction on $m + n$. If $m + n = 0$ (i.e. $m = n = 0$), then the implication holds vacuously. Now, let us assume that the implication holds for all bottom-left flushed $m \times n$ matrix such that $m+n < k$. Consider an arbitrary bottom-left flushed $m \times n$ matrix M such that $m+n = k$. Let $c = cs(\bar{M})$ and $r = rs(M)$. We consider two cases as in the following two figures,



where we have all 0's above the "stairs" (jagged sold lines) and all 1's below the stairs.

Case $M_{mn} = 0$. Since M is bottom-left flushed, the last column of M is all zero like the above left figure. Let M^* be the $m \times (n-1)$ matrix obtained from M by deleting the last column of M. Note that M^* is also bottom-left bottom-left flushed. Let $c^* = cs(\overline{M^*})$ and $r^* = rs(M^*)$. Then

$$r = r^* \text{ and } c = \left(c_1^*, \ldots, c_{n-1}^*, m\right).$$

(In the above, c and c^* are the column sum vectors of $\bar{M}$ and $\overline{M^*}$, respectively, and hence they count the number of 0 on the columns of M and M^*). Thus

$$\{c_1 + 1, \ldots, c_n + n, r_1 + 1, \ldots, r_m + m\}$$

$$
\begin{aligned}
&= \left\{c_1^* + 1, \ldots, c_{n-1}^* + n - 1, m + n, r_1^* + 1, \ldots, r_m^* + m\right\} \\
&= \left\{c_1^* + 1, \ldots, c_{n-1}^* + n - 1, r_1^* + 1, \ldots, r_m^* + m\right\} \cup \{m + n\} \\
&= \{1, \ldots, m + n - 1\} \cup \{m + n\} \qquad \text{from the induction hypothesis} \\
&= \{1, \ldots, m + n\}.
\end{aligned}
$$

Case $M_{mn} = 1$. Since M is bottom-left flushed, the last row of M is all one like the above right figure. Let M^* be the $(m-1) \times n$ matrix obtained from M by deleting the last row of M. Note that M^* is also bottom-left flushed. Let $c^* = cs(\overline{M^*})$ and $r^* = rs(M^*)$. Note that

$$
c = c^* \text{ and } r = \left(r_1^*, \ldots, r_{m-1}^*, n\right).
$$

Thus

$$
\begin{aligned}
&\{c_1 + 1, \ldots, c_n + n, r_1 + 1, \ldots, r_m + m\} \\
&= \left\{c_1^* + 1, \ldots, c_n^* + n, r_1^* + 1, \ldots, r_{m-1}^* + m - 1, n + m\right\} \\
&= \left\{c_1^* + 1, \ldots, c_n^* + n, r_1^* + 1, \ldots, r_{m-1}^* + m - 1\right\} \cup \{n + m\} \\
&= \{1, \ldots, m - 1 + n\} \cup \{n + m\} \quad \text{from the induction hypothesis} \\
&= \{1, \ldots, m + n\}.
\end{aligned}
$$

Therefore, in both cases, $PC(\bar{M}, M)$ holds. □

Definition 10 (Sorted/Flushed)**.** *Let $A, B \in \{0,1\}^{m \times n}$. We say that (A, B) is* sorted *iff $acs(A)$ and $ars(B)$ are sorted in increasing order. We say that (A, B) is* flushed *iff B is bottom-left flushed. We say that (A, B) is* sorted-flushed *iff it is both sorted and flushed.*

Example 10. *Let*

$$
(A, B) = \left[\begin{array}{ccc||c} 1 & 0 & 1 & \\ 0 & 1 & 1 & \\ 0 & 1 & 1 & \\ 1 & 0 & 0 & \\ \hline\hline 2 & 2 & 3 & c_j \\ 3 & 4 & 6 & c_j + j \end{array}\right], \left[\begin{array}{ccc||cc} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 2 \\ 1 & 1 & 0 & 2 & 5 \\ 1 & 1 & 1 & 3 & 7 \\ \hline\hline & & & r_i & r_i + i \end{array}\right]
$$

Note that $acs(A) = (3, 4, 6)$ and $ars(B) = (1, 2, 5, 7)$ are sorted in increasing order. Thus (A, B) is sorted. Note that B is bottom-left flushed. Thus (A, B) is flushed. Hence (A, B) is sorted-flushed.

Lemma 5. *Let $A, B \in \{0,1\}^{m \times n}$. If (A, B) is sorted-flushed, then we have*

$$
cs(A) = cs(\bar{B}) \iff PC(A, B)
$$

Proof. Assume that (A, B) is sorted-flushed. We need to show $cs(A) = cs(\bar{B}) \iff PC(A, B)$. We will show direction of implication one by one.
$\boxed{\Rightarrow}$ Assume $cs(A) = cs(\bar{B})$. Then we have $acs(A) = acs(\bar{B})$. From Lemma 4, we have $PC(\bar{B}, B)$. Thus we have $PC(A, B)$.
$\boxed{\Leftarrow}$ Assume $PC(A, B)$. Then we have

$$
acs(A) = (1, \ldots, m + n) \setminus ars(B).
$$

From Lemma 4, we have $PC(\bar{B}, B)$. Thus we have

$$
acs(\bar{B}) = (1, \ldots, m + n) \setminus ars(B).
$$

Thus $acs(A) = acs(\bar{B})$ and in turn $cs(A) = cs(\bar{B})$. □

Lemma 6. *Let $t = \alpha^{\mu}\beta^{\nu}$ be a term. Let $\mathcal{R} \in \{0,1\}^{m\times n}$. The following two are equivalent.*

(1) $\mathcal{R}$ is a res-representation of t.

(2) $(\mathcal{R}, F_\nu)$ is a sorted-flushed syl-representation of t.

Proof. We show each direction of implication one by one.
(1) ⇒ (2). It follows immediately from the following claims:

C1: *$(\mathcal{R}, F_\nu)$ is sorted-flushed.*

From the definition of F_ν, it is obvious that $ars(F_\nu)$ is sorted in increasing order and that F_ν is bottom-left flushed. Thus it remains to show that $acs(\mathcal{R})$ is sorted in increasing order. Since $\mathcal{R}$ is a res-representation of t, we have $cs(\mathcal{R}) = \bar{\nu}$. Recall that at the very beginning of this section we assumed, without loss of generality, that ν is sorted in non-increasing order. So, $cs(\mathcal{R})$ is sorted in non-decreasing order. Thus, $acs(\mathcal{R})$ is sorted in increasing order.

C2: *$(\mathcal{R}, F_\nu)$ is a syl-representation of t.*

Since $\mathcal{R}$ is a res-representation of t, we have $rs(\mathcal{R}) = \mu$. From the definitions of F_ν, we have $cs(\mathcal{F}_\nu) = \nu$. Thus it remains to show that $PC(\mathcal{R}, F_\nu)$. Note

$$cs(\mathcal{R}) = \bar{\nu} = \overline{cs(F_\nu)} = cs(\overline{F_\nu}).$$

Therefore, from C1 and Lemma 5, we have $PC(\mathcal{R}, F_\nu)$.

(2) ⇒ (1). Since $(\mathcal{R}, F_\nu)$ is a syl-representation of t, we have $rs(\mathcal{R}) = \mu$. Thus it remains to show that $cs(\mathcal{R}) = \bar{\nu}$. Since $(\mathcal{R}, F_\nu)$ is a sorted-flushed syl-representation of t, we have that $(\mathcal{R}, F_\nu)$ is sorted-flushed and $PC(\mathcal{R}, F_\nu)$. Thus, from Lemma 5, we have

$$cs(\mathcal{R}) = cs(\overline{F_\nu}) = \overline{cs(F_\nu)} = \bar{\nu}.$$

□

Algorithm 1 ($SylFromRes$)**.**

In: *$\mathcal{R}$, a res-representation of a term t*

Out: *$(\mathcal{S}_1, \mathcal{S}_2)$, a syl-representation of the term t*

$c \leftarrow cs(\mathcal{R})$

$(\mathcal{S}_1, \mathcal{S}_2) \leftarrow (\mathcal{R}, F_{\bar{c}})$

return $(\mathcal{S}_1, \mathcal{S}_2)$

Example 11. *We trace the algorithm $SylFromRes$ on the following input.*

In: $\mathcal{R} = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$,

which is a res-representation of the term $t = \alpha_1^3\alpha_2^3\alpha_3^3\alpha_4^2\alpha_5^2\beta_1^2\beta_2^2\beta_3^2\beta_4^1$.

$c = [3, 3, 3, 4]$

$$F_{\bar{c}} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

Out: $(\mathcal{S}_1, \mathcal{S}_2) = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix},$

which is a syl-representation of the term t.

Lemma 7. *The algorithm 1 (SylFromRes) is correct. Thus if a term has a res-representation then it has a syl-representation.*

Proof. Let $\mathcal{R}$ be an input, a res-representation of a term $t = \alpha^{\mu}\beta^{\nu}$. Then $c = \bar{\nu}$ and thus $\bar{c} = \nu$. From Lemma 6, $(\mathcal{R}, F_{\bar{c}})$ is a (sorted-flushed) syl-representation of the term t. □

5.4 If a term has a syl-representation then it has a res-representation.

The proof is constructive, that is, it provides an algorithm that takes a syl-representation and produces a res-representation (Algorithm 4). We will again use the key lemma (Lemma 6 from the previous subsection) that establishes a crucial relationship between the two representations (syl and res). In order to use the key lemma, we need to find an algorithm that transforms a given syl-representation into a sorted-flushed syl-representation. We will describe such an algorithm in this subsection. We will divide, naturally, the algorithm into two subalgorithms.

- Algorithm 2 (*Sort*):

  It transforms a syl-representation of a term into a sorted syl-representation of the term. It essentially carries out bubble sort.

- Algorithm 3 (*Flush*):

  It transforms a sorted syl-representation of a term into a sorted-flushed syl-representation of the term. It essentially carries out repeated swapping of entries of the syl-representation to make it flushed while remaining sorted syl-representation.

Most of this subsection will devoted in describing and proving the correctness of the two subalgorithms. Now we plunge into details.

Algorithm 2 (*Sort*)**.**

In: $(\mathcal{S}_1, \mathcal{S}_2)$*, a syl-representation of a term t*

Out: $(\mathcal{S}'_1, \mathcal{S}'_2)$*, a sorted syl-representation of the term t*

1. $(\mathcal{S}'_1, \mathcal{S}'_2) \leftarrow (\mathcal{S}_1, \mathcal{S}_2)$
2. *Repeat*
   (a) $C \leftarrow acs\,(\mathcal{S}'_1)$
   (b) *If C is in increasing order then exit the Repeat loop*

(c) *Find* $j \in \{1, \ldots, n-1\}$ *such that* $C_j > C_{j+1}$

(d) $h \leftarrow C_j - C_{j+1}$

(e) *Repeat* h *times*

i. *Find* $i \in \{1, \ldots, m\}$ *such that* $\mathcal{S}'_{1,i,j} = 1$ *and* $\mathcal{S}'_{1,i,j+1} = 0$

ii. *Swap* $\mathcal{S}'_{1,i,j}$ *and* $\mathcal{S}'_{1,i,j+1}$

3. *Repeat*

(a) $R \leftarrow ars\,(\mathcal{S}'_2)$

(b) *If* R *is in increasing order then exit the Repeat loop*

(c) *Find* $i \in \{1, \ldots, m-1\}$ *such that* $R_i > R_{i+1}$

(d) $h \leftarrow R_i - R_{i+1}$

(e) *Repeat* h *times*

i. *Find* $j \in \{1, \ldots, n\}$ *such that* $\mathcal{S}'_{2,i,j} = 1$ *and* $\mathcal{S}'_{2,\,i+1,\,j} = 0$

ii. *Swap* $\mathcal{S}_{2,i,j}$ *and* $\mathcal{S}_{2,i+1,\,j}$

4. *Return* $(\mathcal{S}'_1, \mathcal{S}'_2)$

Example 12. *We trace the algorithm Sort on the following input.*

In: $(\mathcal{S}_1, \mathcal{S}_2) = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix},$

which is a syl-representation of the term $t = \alpha_1^3\alpha_2^3\alpha_3^3\alpha_4^2\alpha_5^2\beta_1^2\beta_2^2\beta_3^2\beta_4^1$.

1. $(\mathcal{S}'_1, \mathcal{S}'_2) = (\mathcal{S}_1, \mathcal{S}_2)$

2. *Iteration 1*

(a) $C = [2, 6, 8, 7]$

(b) C *is not sorted*

(c) $j = 3$

(d) $h = 1$

i. $i = 3$

ii. *Swap* $S'_{1,3,3}$ *and* $S'_{1,3,4}$

$$\mathcal{S}'_1 = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

Iteration 2

(a) $C = [2, 6, 7, 8]$

(b) C *is sorted in increasing order*

3. Iteration 1

(a) $R = [1, 3, 5, 4, 9]$

(b) R is not sorted

(c) $i = 3$

(d) $h = 1$

i. $j = 1$

ii. Swap $S'_{2,3,1}$ *and* $S'_{2,4,1}$

$$\mathcal{S}'_2 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

Iteration 2

(a) $R = [1, 3, 4, 5, 9]$

(b) R is sorted in increasing order

4. Return $(\mathcal{S}'_1, \mathcal{S}'_2)$

Out: $(\mathcal{S}'_1, \mathcal{S}'_2) = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix},$

which is a sorted syl-representation of the term t.

Lemma 8. *The algorithm 2 (Sort) is correct.*

Proof. Let $(\mathcal{S}_1, \mathcal{S}_2)$ be an input, that is, a syl-representation of a term $t = \alpha^\mu \beta^\nu$. The correctness of the algorithm is immediate from the following claims.

C1: *Right after Step 2,* $(\mathcal{S}'_1, \mathcal{S}'_2)$ *is a syl-representation of the term t and* $acs(\mathcal{S}'_1)$ *is sorted in increasing order.* The proof of the claim is immediate from the following sub-claims.

1. *Right before Step 2a,* $(\mathcal{S}'_1, \mathcal{S}'_2)$ *is a syl-representation of the term t.*
   We prove it by induction on the number of iterations. At the first iteration, it is trivially true since $(\mathcal{S}'_1, \mathcal{S}'_2) = (\mathcal{S}_1, \mathcal{S}_2)$. We assume that it is true after some number of iterations. We need to show that it is still true after one more iteration.
   It is immediate from the following observations.
   - $rs(\mathcal{S}'_1) = \mu$.
     Obvious since the loop body does not change $rs(\mathcal{S}'_1)$.
   - $cs(\mathcal{S}'_2) = \nu$.
     Obvious since the loop body does not change $\mathcal{S}'_2$
   - $PC(\mathcal{S}'_1, \mathcal{S}'_2)$.
     Let $A, B \in \{1, \ldots, m\}$ such that $A = C_j$ and $B = C_{j+1}$. From Step 2d we know that $C_j = C_{j+1} + h$. Inside Step 2e, we have increased C_{j+1} by h and decreased C_j by h . Thus after Step 2e we have

     $$C_{j+1} \leftarrow C_{j+1} + h$$

$$C_j \leftarrow C_j - h$$

Hence $C_j = B$ and $C_{j+1} = A$, that is, we have swapped C_j and C_{j+1}. In other words, the loop body does not change C as a set. Thus $PC\,(\mathcal{S}'_1, \mathcal{S}'_2)$ still holds.

2. *The repeat loop in Step 2 terminates.*
   It is a bubble sort algorithm executed on a finite list C. Therefore it terminates.
3. *In Step 2c, there exists $j \in \{1, \ldots, n-1\}$ such that $C_j > C_{j+1}$.*
   The claim is immediate from the following observations.
   - Since we are at Step 2c, the 'if' condition in Step 2b is not satisfied. Hence C is not in increasing order. Thus there exists $j \in \{1, \ldots, n-1\}$ such that $C_j \geq C_{j+1}$.
   - Since $(\mathcal{S}'_1, \mathcal{S}'_2)$ is a syl-representation, we have $C_j \neq C_{j+1}$.
4. *In Step 2e(i), there exists $i \in \{1, \ldots, m\}$ such that $\mathcal{S}'_{1,i,j} = 1$ and $\mathcal{S}'_{1,i,j+1} = 0$.*
   From Step 2d we know $h = C_j - C_{j+1}$. Hence there must exist h different $i \in \{1, \ldots, m\}$ such that $\mathcal{S}'_{1,i,j} = 1$ and $\mathcal{S}'_{1,i,j+1} = 0$.

C2: *Right after Step 3, $(\mathcal{S}'_1, \mathcal{S}'_2)$ is a syl-representation of the term t and $acs((\mathcal{S}'_1)$ and $ars(\mathcal{S}'_2)$ are sorted in increasing order.* The proof of the claim is symmetric to the proof of C1. One only needs to switch the roles of S'_1 and S'_2 and the roles of columns and rows.

□

Algorithm 3 (*Flush*)**.**

In: *$(\mathcal{S}_1, \mathcal{S}_2)$, a sorted syl-representation of a term t*

Out: *$(\mathcal{S}'_1, \mathcal{S}'_2)$, a sorted-flushed syl-representation of the term t*

1. $(\mathcal{S}'_1, \mathcal{S}'_2) \leftarrow (\mathcal{S}_1, \mathcal{S}_2)$
2. *Repeat*
   - (a) *If $(\mathcal{S}'_1, \mathcal{S}'_2)$ is flushed then return $(\mathcal{S}'_1, \mathcal{S}'_2)$*
   - (b) $c \leftarrow cs(\mathcal{S}'_1)$
     $r \leftarrow rs(\mathcal{S}'_2)$
     $C \leftarrow acs\,(\mathcal{S}'_1)$
     $R \leftarrow ars\,(\mathcal{S}'_2)$
   - (c) *Find $(i,j) \in \{1, \ldots, m-1\} \times \{1, \ldots, n\}$ such that $\mathcal{S}'_{2,i,j} = 1$ and $\mathcal{S}'_{2,i+1,j} = 0$*
     Swap $\mathcal{S}'_{2,i,j}$ and $\mathcal{S}'_{2,i+1,j}$
   - (d) $i_\ell \leftarrow \min\{\, k \mid r_k = r_i,\, k \leq i\}$
   - (e) *If $i_\ell < i$ then*
     Find $j \in \{1, \ldots, n\}$ such that $\mathcal{S}'_{2,i_\ell,j} = 1$ and $\mathcal{S}'_{2,i,j} = 0$
     Swap $\mathcal{S}'_{2,i_\ell,j}$ and $\mathcal{S}'_{2,i,j}$
   - (f) $i_u \leftarrow \max\{\, k \mid r_k = r_{i+1},\, k \geq i+1\}$
   - (g) *If $i+1 < i_u$ then*
     Find $j \in \{1, \ldots, n\}$ such that $\mathcal{S}'_{2,i+1,j} = 1$ and $\mathcal{S}'_{2,i_u,j} = 0$.
     Swap $\mathcal{S}'_{2,i+1,j}$ and $\mathcal{S}'_{2,i_u,j}$
   - (h) *Find $i \in \{1, \ldots, m\}$ and $j_\ell < j_u \in \{1, \ldots, n\}$ such that*
     $\mathcal{S}'_{1,i,j_\ell} = 0$ and $\mathcal{S}'_{1,i,j_u} = 1$ and $C_{j_\ell} = R_{i_\ell} - 1$ and $C_{j_u} = R_{i_u} + 1$
     Swap $\mathcal{S}'_{1,i,j_\ell}$ and $\mathcal{S}'_{1,i,j_u}$

Example 13. *We trace the algorithm Flush on the following input.*

In: $(\mathcal{S}_1, \mathcal{S}_2) = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix},$

which is a sorted syl-representation of the term $t = \alpha_1^3\alpha_2^3\alpha_3^3\alpha_4^2\alpha_5^2\beta_1^2\beta_2^2\beta_3^2\beta_4^1$.

1. $(\mathcal{S}'_1, \mathcal{S}'_2) = (\mathcal{S}_1, \mathcal{S}_2)$
2. *Iteration 1*
   - (a) $(\mathcal{S}'_1, \mathcal{S}'_2)$ *is not flushed*
   - (b) $c = [1, 4, 4, 4]$
     $r = [0, 1, 1, 1, 4]$
     $C = [2, 6, 7, 8]$
     $R = [1, 3, 4, 5, 9]$
   - (c) $i = 2$, $j = 2$
     Swap $\mathcal{S}'_{2,2,2}$ *and* $\mathcal{S}'_{2,3,2}$
     $(\mathcal{S}'_1, \mathcal{S}'_2) = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}$
   - (d) $i_\ell = 2$
   - (e) $i_\ell \not< i$
   - (f) $i_u = 4$
   - (g) $i + 1 < i_u$
     $j = 2$
     Swap $\mathcal{S}'_{2,3,2}$ *and* $\mathcal{S}'_{2,4,2}$
     $(\mathcal{S}'_1, \mathcal{S}'_2) = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}$
   - (h) $i = 1$ *and* $j_\ell = 1$, $j_u = 2$
     Swap $\mathcal{S}'_{1,1,1}$ *and* $\mathcal{S}'_{1,1,2}$
     $(\mathcal{S}'_1, \mathcal{S}'_2) = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}$

   Iteration 2

   - (a) $(\mathcal{S}'_1, \mathcal{S}'_2)$ *is not flushed*

(b) $c = [2, 3, 4, 4]$
$r = [0, 0, 1, 2, 4]$
$C = [3, 5, 7, 8]$
$R = [1, 2, 4, 6, 9]$

(c) $i = 3$, $j = 3$
Swap $\mathcal{S}'_{2,3,3}$ *and* $\mathcal{S}'_{2,4,3}$

$$(\mathcal{S}'_1, \mathcal{S}'_2) = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

(d) $i_\ell = 3$

(e) $i_\ell \not< i$

(f) $i_u = 4$

(g) $i + 1 \not< i_u$

(h) $i = 2$ *and* $j_\ell = 1$, $j_u = 3$
Swap $\mathcal{S}'_{1,2,1}$ *and* $\mathcal{S}'_{1,2,3}$

$$(\mathcal{S}'_1, \mathcal{S}'_2) = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

Iteration 3

(a) $(\mathcal{S}'_1, \mathcal{S}'_2)$ *is flushed*

Out: $(\mathcal{S}'_1, \mathcal{S}'_2) = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix},$

which is a sorted-flushed syl-representation of the term t.

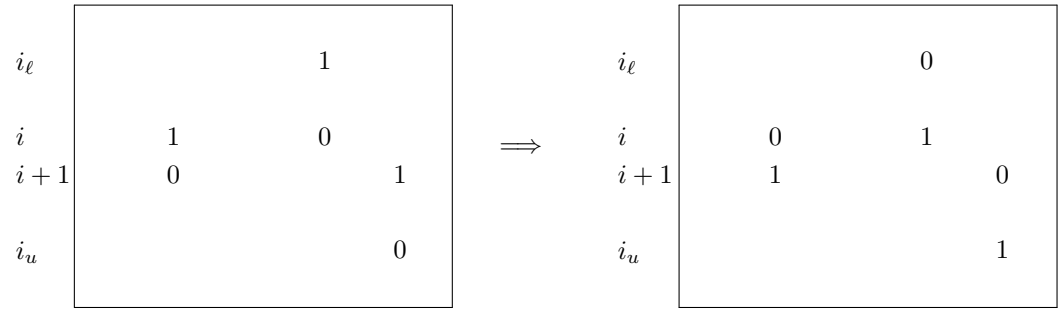
Lemma 9. *The algorithm 3* ($Flush$) *is correct.*

Proof. Let $(\mathcal{S}_1, \mathcal{S}_2)$ be an input, that is, a sorted syl-representation of a term $t = \alpha^\mu \beta^\nu$. The correctness of the algorithm is immediate from the following claims.

C1: *Right before Step 2a,* $(\mathcal{S}'_1, \mathcal{S}'_2)$ *is a sorted syl-representation of the term* t.

We prove it by induction on the number of iterations. At the first iteration, it is trivially true since $(\mathcal{S}'_1, \mathcal{S}'_2) = (\mathcal{S}_1, \mathcal{S}_2)$. We assume that it is true after some number of iterations. We need to show that it is still true after one more iteration. It is immediate from the following two sub-claims, where $C' = acs(\mathcal{S}'_1)$ and $R' = ars(\mathcal{S}'_2)$ at the end of Step 2h.

- C' *and* R' *are sorted in increasing order.*
  Note that Steps 2c–2g transform $\mathcal{S}'_2$ by carrying out swaps along columns, as depicted by the following diagram.



Thus

$$R'_t = \begin{cases} R_t - 1 & \text{if} \quad t = i_\ell \\ R_t + 1 & \text{if} \quad t = i_u \\ R_t & \text{else} \end{cases}$$

From Step 2h, it is immediate that

$$C'_t = \begin{cases} C_t + 1 & \text{if} \quad t = j_\ell \\ C_t - 1 & \text{if} \quad t = j_u \\ C_t & \text{else} \end{cases}$$

Next recall that R and C are sorted in increasing order. Thus we only need to show that $R'_{i_\ell} > R_{i_\ell - 1}$, $R'_{i_u} < R_{i_u+1}$, $C'_{j_\ell} < C_{j_\ell+1}$ and $C'_{j_u} > C_{j_u-1}$. We show them one by one.

- $R'_{i_\ell} > R_{i_\ell-1}$
  Recall that $R'_{i_\ell} = R_{i_\ell} - 1$. Since $r_{i_\ell} > r_{i_\ell-1}$ we have $R_{i_\ell} - 1 > R_{i_\ell-1}$. Thus $R'_{i_\ell} > R_{i_\ell-1}$.
- $R'_{i_u} < R_{i_u+1}$
  Recall that $R'_{i_u} = R_{i_u} + 1$. Since $r_{i_u} < r_{i_u+1}$ we have $R_{i_u} + 1 < R_{i_u+1}$. Thus $R'_{iu} < R_{i_u+1}$.
- $C'_{j_\ell} < C_{j_\ell+1}$
  Recall that $C'_{j_\ell} = C_{j_\ell} + 1$. Since $C_{j_\ell} = R_{i_\ell} - 1$ we have $C'_{j_\ell} = R_{i_\ell}$. Since R_{i_ℓ} appears in R and $PC\,(C, R)$, we see that R_{i_ℓ} does not appear in C. Hence $R_{i_\ell} < C_{j_\ell+1}$. Thus $C'_{j_\ell} < C_{j_\ell+1}$.
- $C'_{j_u} > C_{j_u-1}$
  Recall that $C'_{j_u} = C_{j_u} - 1$. Since $C_{j_u} = R_{i_u} + 1$ we have $C'_{j_u} = R_{i_u}$. Since R_{i_u} appears in R and $PC\,(C, R)$, we see that R_{i_u} does not appear in C. Hence $R_{i_u} > C_{j_u-1}$. Thus $C'_{j_u} > C_{j_u-1}$.

- *$(\mathcal{S}'_1, \mathcal{S}'_2)$ is a syl-representation of the term t at the end of Step 2h.*
  - $rs\,(\mathcal{S}'_1) = \mu$.
    Obvious since the loop body does not change $rs\,(\mathcal{S}'_1)$.
  - $cs\,(\mathcal{S}'_2) = \nu$.
    Obvious since the loop body does not change $cs\,(\mathcal{S}'_2)$.
  - $PC\,(\mathcal{S}'_1, \mathcal{S}'_2)$ holds.
    Note $R'_{i_\ell} = C_{j_\ell}$, $R'_{i_u} = C_{j_u}$ and $C'_{j_\ell} = R_{i_\ell}$, $C'_{j_u} = R_{i_u}$. Note that the others do not change. In other words $C' \cup R' = C \cup R$ as sets. Thus $PC\,(\mathcal{S}'_1, \mathcal{S}'_2)$ holds.

C2: *The main loop (Repeat) terminates.*

For every iteration of the main loop, at least one "out of order" pair of $(1, 0)$ in $\mathcal{S}_2$ is swapped. Thus the algorithm terminates.

C3: *In Step 2c, there exists (i,j) satisfying the conditions stated in the step.*

Since we are at Step 2c, the 'if' condition in Step 2a is not satisfied. Hence $(\mathcal{S}'_1, \mathcal{S}'_2)$ is not flushed. Thus there exists $(i, j) \in \{1, \ldots, m-1\} \times \{1, \ldots, n\}$ such that $\mathcal{S}'_{2,i,j} = 1$ and $\mathcal{S}'_{2,i+1,j} = 0$.

C4: *In Step 2e, there exists j satisfying the conditions stated in the step.*

Let $r' = rs(\mathcal{S}'_2)$ right before entering the step. Note

- From Step 2c, we have $r'_i = r_i - 1$.
- From Step 2d, we have $r'_{i_\ell} = r_{i_\ell} = r_i$.

Therefore there exist $j \in \{1, \ldots, n\}$ such that $\mathcal{S}'_{2,i,j} = 0$ and $\mathcal{S}'_{2,i_\ell,j} = 1$.

C5: *In Step 2g, there exists j satisfying the conditions stated in the step.*

Let $r' = rs(\mathcal{S}'_2)$ right before entering the step. Note

- From Step 2c, we have $r'_{i+1} = r_{i+1} + 1$.
- From Step 2f, we have $r'_{i_u} = r_{i_u} = r_{i+1}$.

Therefore there exist $j \in \{1, \ldots, n\}$ such that $\mathcal{S}'_{2,i+1,j} = 1$ and $\mathcal{S}'_{2,i_u,j} = 0$.

C6: *In Step 2h, there exist i, j_ℓ, j_u satisfying the conditions stated in the step.*

1. From Step 2d, we have
   - if $i_\ell = 1$ then $R_{i_\ell} \geq 2$ (since $\mathcal{S}'_{2,i,j} = 1$ and in turn $r_{i_\ell} = r_i \geq 1$).
   - if $i_\ell > 1$ then $R_{i_\ell} - R_{i_\ell - 1} \geq 2$ (since $r_{i_\ell} \geq r_{i_\ell - 1} + 1$).

   Thus $R_{i_\ell} - 1$ does not appear in R. Hence it must appear in C. Thus there exists j_ℓ such that $C_{j_\ell} = R_{i_\ell} - 1$.
2. From Step 2f, we have
   - if $i_u = m$ then $R_{i_u} \leq m + n - 1$ (since $\mathcal{S}'_{2,i+1,j} = 0$ and in turn $r_{i_u} = r_{i+1} \leq n - 1$).
   - if $i_u < m$ then $R_{i_u+1} - R_{i_u} \geq 2$ (since $r_{i_u} + 1 \leq r_{i_u+1}$).

   Thus $R_{i_u} + 1$ does not appear in R. Hence it must appear in C. Thus there exists j_u such that $C_{j_u} = R_{i_u} + 1$.

Therefore there exist $j_\ell, j_u \in \{1, \ldots, n\}$ such that $C_{j_\ell} = R_{i_\ell} - 1$ and $C_{j_u} = R_{i_u} + 1$.

It remains to show that there exists i that satisfies the conditions of Step 2h.

Note that $R_{i_\ell}, R_{i_\ell+1}, \ldots, R_{i_u}$ appear in R. Hence they do not appear in C. Note that

$$\begin{aligned}
R_{i_\ell} &= r_i + i_\ell \\
R_{i_\ell+1} &= r_i + i_\ell + 1 \\
&\vdots \\
R_i &= r_i + i \\
R_{i+1} &= r_{i+1} + i + 1 \\
R_{i+2} &= r_{i+1} + i + 2 \\
&\vdots \\
R_{i_u} &= r_{i+1} + i_u
\end{aligned}$$

Note that $R_{i_\ell}, \ldots, R_i$ are consecutive integers. Likewise note that $R_{i+1}, \ldots, R_{i_u}$ are consecutive integers. We show that $j_u - j_\ell = R_{i+1} - R_i$. Consider the following two cases:

Case 1: $R_i + 1 = R_{i+1}$

Note that $R_{i_\ell}, \ldots, R_{i_u}$ are consecutive and they do not appear in C. Since C is sorted in increasing order and $C_{j_\ell} = R_{i_\ell} - 1$ and $C_{j_u} = R_{i_u} + 1$, there is nothing in between C_{j_ℓ} and C_{j_u}. Hence $j_u - j_\ell = 1$. Note that $R_{i+1} - R_i = 1$. Thus $j_u - j_\ell = R_{i+1} - R_i$.

Case 2: $R_i + 1 < R_{i+1}$

Note that the consecutive list of numbers $R_i + 1, \ldots, R_{i+1} - 1$ do not appear in R. Hence they appear in C. Since C is sorted in increasing order and $C_{j_\ell} = R_{i_\ell} - 1$ and $C_{j_u} = R_{i_u} + 1$, we conclude that exactly $R_i + 1, \ldots, R_{i+1} - 1$ appear in between C_{j_ℓ} and C_{j_u}. Hence

$$j_u - j_\ell - 1 = (R_{i+1} - 1) - (R_i + 1) + 1 = R_{i+1} - R_i - 1$$

Thus $j_u - j_\ell = R_{i+1} - R_i$.

In both cases, we have shown that $j_u - j_\ell = R_{i+1} - R_i$. Note

$$\begin{aligned} c_{j_u} - c_{j_\ell} &= (C_{j_u} - j_u) - (C_{j_\ell} - j_\ell) \\ &= (C_{j_u} - C_{j_\ell}) - (j_u - j_\ell) \\ &= ((R_{i_u} + 1) - (R_{i_\ell} - 1)) - (R_{i+1} - R_i) \\ &= ((r_{i+1} + i_u + 1) - (r_i + i_\ell - 1)) - ((r_{i+1} + i + 1) - (r_i + i)) \\ &= i_u - i_\ell + 1 \\ &\geq 2 \end{aligned}$$

Hence there exists $i \in \{1, \ldots, m\}$ such that $\mathcal{S}'_{1,i,j_\ell} = 0$ and $\mathcal{S}'_{1,i,j_u} = 1$.

□

Algorithm 4 (*ResFromSyl*)**.**

In: $(\mathcal{S}_1, \mathcal{S}_2)$, *a syl-representation of a term* t

Out: $\mathcal{R}$, *a res-representation of the term* t

1. $(\mathcal{S}'_1, \mathcal{S}'_2) \leftarrow Sort\,(\mathcal{S}_1, \mathcal{S}_2)$
2. $(\mathcal{S}'_1, \mathcal{S}'_2) \leftarrow Flush\,(\mathcal{S}'_1, \mathcal{S}'_2)$
3. $\mathcal{R} \leftarrow \mathcal{S}'_1$
4. *Return* $\mathcal{R}$.

Example 14. *We trace the algorithm ResFromSyl on the following input.*

In: $(\mathcal{S}_1, \mathcal{S}_2) = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix},$

which is a syl-representation of the term $t = \alpha_1^3\alpha_2^3\alpha_3^3\alpha_4^2\alpha_5^2\beta_1^2\beta_2^2\beta_3^2\beta_4^1$.

1. $(\mathcal{S}'_1, \mathcal{S}'_2) = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}$

2. $(\mathcal{S}_1', \mathcal{S}_2') = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}$

3. $\mathcal{R} = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$

Out: $\mathcal{R} = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$,

which is a res–representation of the term t.

Lemma 10. *The algorithm 4 ($ResFromSyl$) is correct. Thus if a term has a syl-representation then it has a res-representation.*

Proof. Let $(\mathcal{S}_1, \mathcal{S}_2)$ be an input, that is, a syl-representation of a term t. The correctness of the algorithm is immediate from the following claims.

C1: *After Step 1, $(\mathcal{S}_1', \mathcal{S}_2')$ is a sorted syl representation of the term t.*

Immediate from the specification of Algorithm 2 ($Sort$) and Lemma 8.

C2: *After Step 2, $(\mathcal{S}_1', \mathcal{S}_2')$ is a sorted-flushed syl-representation of the term t.*

Immediate from C1, the specification of Algorithm 3 ($Flush$) and Lemma 9.

C3: *After Step 3, $\mathcal{R}$ is a res representation of the term t*

Immediate from C2 and Lemma 6.

□

References

[1] M. F. Atiyah and I. G. Macdonald. *Introduction to commutative algebra*. Addison-Wesley Publishing Co., Reading, Mass.-London-Don Mills, Ont., 1969.

[2] L. Bernardin, P. Chin, P. DeMarco, K. O. Geddes, D. E. G. Hare, K. M. Heal, G. Labahn, J. P. May, J. McCarron, M. B. Monagan, D. Ohashi, and S. M. Vorkoetter. *Maple Programming Guide*. Maplesoft, a division of Waterloo Maple Inc., 2011-2015.

[3] T. Bogart, A. N. Jensen, D. Speyer, B. Sturmfels, and R. R. Thomas. Computing tropical varieties. *J. Symbolic Comput.*, 42(1-2):54–73, 2007.

[4] T. Bogart, A. N. Jensen, D. Speyer, B. Sturmfels, and R. R. Thomas. Computing tropical varieties. *J. Symbolic Comput.*, 42(1-2):54–73, 2007.

[5] Erwan Brugallé and Kristin Shaw. A bit of tropical geometry. *Amer. Math. Monthly*, 121(7):563–589, 2014.

[6] Bruno Buchberger. *An Algorithm for Finding the Basis Elements of the Residue Class Ring of a Zero Dimensional Polynomial Ideal*. PhD thesis, University of Innsbruck, 1965.

[7] Bruno Buchberger. An algorithm for finding the basis elements of the residue class ring of a zero dimensional polynomial ideal. *J. Symbolic Comput.*, 41(3-4):475–511, 2006. Translated from the 1965 German original by Michael P. Abramson.

[8] Peter Butkovič. *Max-linear systems: theory and algorithms*. Springer Monographs in Mathematics. Springer-Verlag London, Ltd., London, 2010.

[9] David Cox, John Little, and Donal O'Shea. *Ideals, varieties, and algorithms*. Undergraduate Texts in Mathematics. Springer-Verlag, New York, second edition, 1997. An introduction to computational algebraic geometry and commutative algebra.

[10] Carlos D'Andrea, Hoon Hong, Teresa Krick, and Agnes Szanto. Sylvester's double sums: the general case. *J. Symbolic Comput.*, 44(9):1164–1175, 2009.

[11] Alicia Dickenstein, Eva Maria Feichtner, and Bernd Sturmfels. Tropical discriminants. *J. Amer. Math. Soc.*, 20(4):1111–1133, 2007.

[12] Kazimierz Głazek. *A guide to the literature on semirings and their applications in mathematics and information sciences*. Kluwer Academic Publishers, Dordrecht, 2002. With complete bibliography.

[13] Jonathan S. Golan. *Semirings and their applications*. Kluwer Academic Publishers, Dordrecht, 1999. Updated and expanded version of ıt The theory of semirings, with applications to mathematics and theoretical computer science [Longman Sci. Tech., Harlow, 1992; MR1163371 (93b:16085)].

[14] Jonathan S. Golan. *Semirings and affine equations over them: theory and applications*, volume 556 of *Mathematics and its Applications*. Kluwer Academic Publishers Group, Dordrecht, 2003.

[15] Michel Gondran and Michel Minoux. *Graphs, dioids and semirings*, volume 41 of *Operations Research/Computer Science Interfaces Series*. Springer, New York, 2008. New models and algorithms.

[16] Jeremy Gunawardena, editor. *Idempotency*, volume 11 of *Publications of the Newton Institute*. Cambridge University Press, Cambridge, 1998. Papers from the workshop held in Bristol, October 3–7, 1994.

[17] Jeremy Gunawardena. An introduction to idempotency. In *Idempotency (Bristol, 1994)*, volume 11 of *Publ. Newton Inst.*, pages 1–49. Cambridge Univ. Press, Cambridge, 1998.

[18] Bernd Heidergott, Geert Jan Oldser, and Jacob van der Woude. *Max plus at work*. Princeton Series in Applied Mathematics. Princeton University Press, Princeton, NJ, 2006. Modeling and analysis of synchronized systems: a course on max-plus algebra and its applications.

[19] Ilia Itenberg, Grigory Mikhalkin, and Eugenii Shustin. *Tropical algebraic geometry*, volume 35 of *Oberwolfach Seminars*. Birkhäuser Verlag, Basel, second edition, 2009.

[20] Z. Izhakian and L. Rowen, Supertropical Algebra, Advances in Mathematics 225(2010) 2222-2286.

[21] Z. Izhakian and L. Rowen, Supertropical polynomials and resultants, Journal of Algebra, 324(2010) 1860-1886.

[22] Anders Jensen and Josephine Yu. Computing tropical resultants. *J. Algebra*, 387:287–319, 2013.

[23] Vassili N. Kolokoltsov and Victor P. Maslov. *Idempotent analysis and its applications*, volume 401 of *Mathematics and its Applications*. Kluwer Academic Publishers Group, Dordrecht, 1997. Translation of ıt Idempotent analysis and its application in optimal control (Russian), "Nauka" Moscow, 1994 [MR1375021 (97d:49031)], Translated by V. E. Nazaikinskii, With an appendix by Pierre Del Moral.

[24] Ernst Kunz. *Introduction to commutative algebra and algebraic geometry*. Modern Birkhäuser Classics. Birkhäuser/Springer, New York, 2013. Translated from the 1980 German original [MR0562105] by Michael Ackerman, With a preface by David Mumford, Reprint of the 1985 edition [MR0789602].

[25] G. L. Litvinov and S. N. Sergeev, editors. *Tropical and idempotent mathematics and applications*, volume 616 of *Contemporary Mathematics*. American Mathematical Society, Providence, RI, 2014. Papers from the International Workshop on Tropical and Idempotent Mathematics (Tropical-12) held at the Independent University, Moscow, August 26–31, 2012.

[26] R. Loos. Generalized polynomial remainder sequences. In *Computer algebra*, pages 115–137. Springer, Vienna, 1983.

[27] Diane MacLagan and Bernd Sturmfels. *Introduction to Tropical Geometry*, volume 161 of *Graduate Studies in Mathematics*. American Mathematical Society, 2015.

[28] Maplesoft, a division of Waterloo Maple Inc. *Maple User Manual*, 2005-2015.

[29] Grigory Mikhalkin. Tropical geometry and its applications. In *International Congress of Mathematicians. Vol. II*, pages 827–852. Eur. Math. Soc., Zürich, 2006.

[30] Shinsuke Odagiri. The tropical resultant. *Proc. Japan Acad. Ser. A Math. Sci.*, 84(7):93–96, 2008.

[31] Sakiko Ôhashi. On definition for commutative idempotent semirings. *Proc. Japan Acad.*, 46:113–115, 1970.

[32] Lior Pachter and Bernd Sturmfels. Tropical geometry of statistical models. *Proc. Natl. Acad. Sci. USA*, 101(46):16132–16137 (electronic), 2004.

[33] Jean-Eric Pin. Tropical semirings. *Idempotency*. (Bristol, 1994), 5069, Publ. Newton Inst., Vol. 11, Cambridge Univ. Press, Cambridge, 1998.

[34] George Salmon. *Lessons introductory to the modern higher algebra*. Forth edition. Hodges, Figgis, and Co., Dublin, 1885.

[35] Imre Simon. Recognizable sets with multiplicities in the tropical semiring. *Lecture Notes in Computer Science*, 324:107–120, 1988.

[36] David Speyer and Bernd Sturmfels. Tropical mathematics. *Math. Mag.*, 82(3):163–173, 2009.

[37] Bernd Sturmfels and Ngoc Mai Tran. Combinatorial types of tropical eigenvectors. *Bull. Lond. Math. Soc.*, 45(1):27–36, 2013.

[38] James Joseph Sylvester. On a theory of the syzygetic relations of two rational integral functions, comprising an application to the theory of sturm's functions, and that of the greatest algebraical common measure. *Phil. Trans. R. Soc. Lond.*, 143(1):407–548, 1853.

[39] Luis Felipe Tabera. Tropical resultants for curves and stable intersection. *Rev. Mat. Iberoam.*, 24(3):941–961, 2008.